SECOND GENERATION VIDEO CODING SCHEMES AND THEIR ROLE IN MPEG-4

Luis Torres

Dept. of Signal Theory and Communications
Universitat Politècnica de Catalunya
Barcelona, Spain
E-mail: luis@gps.tsc.upc.es

Murat Kunt

Signal Processing Laboratory
Swiss Federal Institute of Technology
Lausanne, Switzerland
E-mail: kunt@epfl.ch

Fernando Pereira

Instituto Superior Técnico
Department of Electrical and Computer Engineering
Lisboa, Portugal
E-mail: eferbper@beta.ist.utl.pt

ABSTRACT

Since its introduction in 1985, there has been a lot of activity in the field of second generation still image coding. In the last years, the approach has been extended to video coding and has been the basis for European RACE research projects. The objective of this paper is to summarize the main concepts underlying second generation video coding and to present the approach taken in the RACE project MORPHECO in the context of segmentation-based video coding schemes. Comments and results on the evolution of the system, in the framework of the MPEG-4 SESAME proposal presented on behalf of the RACE project MAVT are also outlined. In order to have a more complete paper, current state of the art in MPEG-4 activities are also summarized[1]. The paper ends with conclusions and new trends in the field. A basic introduction to the main characteristics and limitations of first generation coding will be also presented in order to have a self-contained paper. A more complete description of second generation video coding and in particular of the approach presented here can be found in[34] .

KEYWORDS: Video coding, Second generation schemes, Segmentation-based video coding schemes, MPEG-4.

[1]This part of the work has benefited by the Acción Integrada of the Spanish and Portuguese governments: 90 B and E-54/95 respectively

1 INTRODUCTION

The digital representation of an image or of a sequence of images requires a very large number of bits. The goal of image coding is to reduce this number as much as possible, and to reconstruct a faithful duplicate of the original picture. For many years waveform-based image coding approaches have been the only approaches utilized in image compression. One of the the main problems of the now so-called *first generation* coding techniques, is that they did not question the image representation structure based on the concept of pixel or block of pixels as the basic entities that are coded (*the canonical representation of the image*). In addition, they also share in common the absence of consideration for the human visual system (HVS) in the design of the coder.

At the middle of the eighties was quite clear that the basic philosophy behind first generation techniques had reached a saturation[23] and new and innovative techniques were needed. Although at that time video coding schemes were in their infancy, especially approaches dealing with digital television, the techniques developed some years later in the framework of the standards JPEG[15] , H.261[12] and MPEG[21,22] , confirmed that the codecs based in the so called first generation approaches presented limitations for very low bit-rate video coding applications (bit-rates below 64 kbits/s). In 1995 the new standard for very low bit-rate video coding H.263[13] was defined, although it is mainly intended for conversational services and it is not generic.

In order to overcome the limitations imposed by first generation image coding techniques, *second generation* image coding was formally introduced in 1985[17] . We will widely call second generation image and video coding techniques the ensemble of approaches proposing different and alternative image representations than the conventional canonical form, based in the human visual system. The human visual system becomes a fundamental part of the encoding/decoding chain in this new type of representation. As a result of including the human visual system, second generation can be more specifically defined as an approach of seeing the image composed by different entities called objects. This implies that the image or sequence of images have first to be analyzed and/or segmented in order to find the entities.

This paper is organized as follows. Section 2 presents a basic overview of the main limitations of the canonical representation of the image. Section 3 gives some remarks on first generation that help to understand the evolution towards second generation. Section 4 presents the underlying general concepts of second generation techniques and makes a brief overview of the topics in which the MORPHECO scheme is based with special emphasis in segmentation-based approaches. Section 5 presents the general structure of the MORPHECO scheme and shows some examples of encoded sequences using the scheme. Section 6 briefly explains the concepts of dynamic coding in which the SESAME proposal is based. Section 7 presents the evolution of the MORPHECO scheme that has materialized into the MPEG-4 SESAME proposal and shows results of the scheme. Section 8 presents the state of the art in the context of MPEG-4 and describes how second generation techniques can also be applied in this context. Finally, Section 9 presents some conclusions and gives new trends in the field.

2 THE PROBLEMS OF THE CANONICAL REPRESENTATION OF IMAGES

The first problem one faces in compressing visual information is that of representation. How do we represent visual information that each one of us has around us in the four dimensional (4-D) space we are living (3 space and 1 time dimensions)?

The first answer came in the late forties, thanks to the television. The 4-D space is projected on the image plane of a sensor at a regular pace producing a series of single images. An image is scanned line by line and an electrical signal representing the brightness of the scanned area is generated. This analogue signal is then formatted and modulated for transmission. Although this system has worked quite successfully since more than

a half century, it has a number of drawbacks. The analogue 4-D space is projected on a plane. Even if this projection could be made continuous in time, one dimension (and hence information) is lost by projection. In practice, the projection cannot be made continous in time for technical reasons (the image plane need to be scanned). This introduces a first sampling, the sampling in time of the 4-D space. Then, the image plane is sampled vertically to produce the video signal. This is a second sampling in space. We all know, since the early days of the kindergarten, that whenever a sampling operation is performed, attention must be payed to the conditions of the sampling theorem. None of these samplings do that. As a consequence, we define two sets of input information. One set is that of these signals which respect the sampling theorem and can go through the system. The other set, complementary, contains signals that do not respect the theorem. These are necessarily distorted when processed. Typical examples are the wheels of a car going to the right, turning in the aberrant direction as if the car was going to the left, or the material of dresses some TV speakers wear inducing moire patterns and motion though staying still.

In the fully digital world in which we are living today, a third sampling is introduced along the scanning direction of the video signal. It is as careless as the first two with regards to the sampling theorem. Notice that all these three samplings are implemented at a constant sampling rate or period. To be fully digital, all these samples are quantized using a given number of bits per sample. Eventually, we have an enormous set of bits or bit rate that represents our 4-D world. Ironically, we call this representation canonical, implying that we cannot do any better. In fact, we cannot do any worse! The criticism we can formulate may be listed as follows. The 4-D information is processed in the same way we process stationary 1-D signals. Visual information is everything but stationary. Not only do we use sampling without respect for its theorem, but we use practical values for sampling rates that have nothing to do with the input signal. The famous 25 or 30 frames per second rates are dictated not by the input signals but by the frequency of the voltage power line. Almost nothing or very little is done, noticing that the last element of the processing chain is not the display device but the human visual system, with its marvelously rich and poor features. Because we are unable to do adaptive sampling, because we need easily implementable systems and we are still doing our best, we obtain a canonical representation which may be usable for a subset of visual information. Even within this subset, the system may have the right parameters for some part of the information but, due to the nonstationarities, has the wrong parameters for the other parts. As a consequence, the canonical form is not canonical but dramatically redundant. One last comment that can be made is that the structure imposed by the system is not data driven. It is a fully arbitrary structure, independent from the data. Assuming a binary representation of the canonical form, if we were able to generate images representing all the possible combinations of these bits, most of the results would look like anything but visual information. Natural images are only a very small subset. The problem at hand is compressing the canonical form. Viewed from the previous angle it looks like Sysiphus and his rock. It keeps us busy and happy.

The set of compression methods developed till the mid eighties did not question the structure imposed by the canonical representation and tried to combine picture samples in various ways to obtain a compressed bit stream. Natural images contain man made or nature made objects defined not by a set of pixels regularly spaced in all dimensions but by their shape, and color. Even the pointillism was a way for some painters to express their view, a painter never paints by pixels. Shape, texture and color are the most frequently used features. So why not to imitate painters by technical means? As Sysiphus, let's push our rock up hill and try to extract these features from the available data. Computer vision has been trying to implement these operations for quite some time with more or less success. However the constraint is different. In image compression we need to recover a faithfull replica of the original scene, whereas computer vision aims at extracting semantical information or descriptions.

Efforts directed in these new directions developed the so-called second generation techniques for image and video compression.

3 FIRST GENERATION VIDEO CODING TECHNIQUES

In order to better understand the new concepts introduced in second generation video coding schemes, it is important to review very basically the principles in which the coding techniques of the first generation are established. We present in this section a very brief summary of the concepts of first generation schemes that provide insight in the second generation. The basic schemes that are mainly considered first generation are pulse code modulation, predictive coding, transform coding, vector quantization and the myriad of schemes including combinations or particular cases of those such us subband and wavelets. For a more complete information and additional references on these coding schemes, the reader is referred to[2] .

First generation image and video coding techniques achieve compression by reducing the statistical redundancy and the irrelevancy of the image data. To that end, the image is considered as a set of pixels that have to be uncorrelated. This objective is accomplished using waveform coding. The problem is stated in the well known field of rate-distortion theory to achieve the minimum possible waveform distortion for a given coding rate or, equivalently, to achieve a given acceptable level of waveform distortion at the least possible encoding rate[32] . This approach requires the knowledge of the source distribution function and an adequate definition of the distortion measure. A mean square distortion measure has generally been used. This approach does not pay any attention to the semantic meaning of the selected set or block of pixels. This model has been the basis of all first generation schemes and has been very popular in the last twenty years or so. But the last years have seen some sort of concern regarding this model, due mainly to the following general reasons:

- All the attempts to provide a true knowledge of the source distribution function have had a very limited success and only simple models are used.
- It has been quite difficult to incorporate the most important part of a video coding system, the human visual system, into the model. The widely used mean square distortion measure keeps little relation with the human visual system.
- First generation video coding techniques have not proven by themselves useful to cope simultaneously with the requirements of very low bit-rates and the semantic understanding of the images.

One of the main achievements coming from the first generation is the hybrid scheme formed by combining motion compensated prediction in the temporal domain and a decorrelation technique in the spatial domain. This scheme is used in current video standards[21,22,12,13] and may serve as a starting point to introduce some of the basic concepts of the second generation approach. In this scheme the input image is divided into square blocks of usually 8x8 or 16x16 pixels. Three main problems arise when considering this structure:

- The blind division, without taking into account the semantic content of the image, results in block effect when high compression ratios are desired.
- Motion models are applied to square blocks of pixels that may have little resemblance to the true motion of the objects that form the image.
- The properties of the human visual system are not taken into account.

The main conclusions of the actual video coding schemes of the first generation in the context of very low bit-rate video coding can be stated as follows:

- The image deteriorates at high compression ratios, mainly due to the block effect appearance in the decoded image.

- The achievable bit rates range from 8 to 36 kbits/s depending on the difficulty of the video sequence and the desired quality. 8 kbits/s is only achievable for very steady head and shoulders sequences while it seems that bit rates around 24 kbits/s may be reached for more complicated sequences.
- The fact that these techniques are still being applied in a block-based approach will still establish an upper bound, at least in some other fields related to image and video coding, such as content-based multimedia data access or content-based scalability, as defined in the proposal package description of the future standard MPEG-4[16] .
- But it has to be recognized that, in spite of their limitations, first generation video coding techniques share a set of properties that have opened the possibility of a new world in image understanding and manipulation. In particular, texture and motion representation using a block-based approach, have paved the way for a better understanding of their region-based counterparts.

One may wonder if this hybrid scheme is capable of improvements that may give further compression and/or achieve new functionalities. Two main stages can be improved: the motion analysis and the intra-frame spatial decorrelation stages. The reader is referred to[8] for a good discussion on the fact that very accurate motion compensation may not be the key to a better picture quality due to the severe rate-constriction of very low bit-rate coders. Further discussions on the limits and improvements of the hybrid scheme of first generation techniques are presented in[18] .

As an important point with respect to first generation techniques, the new concept of content-based functionalities has to be emphasized. Functionalities have been described as the capability to access and manipulate the image content[16] . It seems that due to the inherent block based nature of the first generation schemes the amount and quality of the functionalities that can be implemented is limited. More details on the topic of functionalities is presented in Section 8.

We would like to conclude this section by noting that first generation schemes are excellent *texture coding schemes*. But, as we mentioned before, first generation video coding techniques have not proven by themselves useful to cope simultaneously with the requirements of very low bit-rates and the semantic understanding of the images. In this context it seems appropriate to explore new compression methods that may overcome the limitations of first generation techniques and may provide additional functionalities. The rest of the paper is dedicated to the so called second generation image and video coding techniques with the objective in mind of providing both requirements.

4 SECOND GENERATION VIDEO CODING TECHNIQUES

In 1985 when the first original paper was published on second generation image coding techniques[17] , it was clear what this concept was. Now, 10 years later, the field of image and video coding has seen a tremendous explosion of new ideas, techniques, implementations and standards. It then seems adequate to briefly review the main concepts in which the techniques belonging to second generation are founded.

We have seen in Section 3 that first generation techniques are based in the framework of waveform coding. We have already seen how this framework has been implemented and the problems that arise for very low bit-rate video coding applications. The basic idea underlying second generation is to overcome these limitations. How can it be done? One possible answer is to incorporate the properties of the human visual system, that, at the end, is the most important part of the coder/decoder chain. How these properties can be incorporated is not an easy task. The first thing to do is to have a detailed knowledge of the HVS. Then, the second thing is to find an adequate model for the image that fulfills this knowledge. If we know that the human visual system is able to recognize an image or a sequence of images using only a very few information points, then the task reduces to

finding the points that convey most of the information. If we want not only to recognize but also to appreciate more detailed information, then we should also be able to incorporate this characteristic into the model. It is adequate in this context to summarize very briefly the main characteristics of the HVS that have led to the introduction of new image and video coding techniques.

4.1 The human visual system

If we were to summarize the three main results concerning the human visual system that are of interest to understand the proposal of second generation image and video coding techniques, we would select the following:

- The edge and contour information are very much appreciated by the human visual system and are responsible for our perception.
- The texture information has relative importance. Texture is associated with *additional* information. It influences our perception when taken together with the contour information.
- The images of many natural objects can be approximated by members of a class of deterministically self-similar sets.

These conclusions explain the reasons that have led to the proposition of different alternatives concerning the integration of the human visual system into the coder design. First, is the proposal of a contour-texture approach to image coding. This has been considered in the MORPHECO project under the scope of segmentation-based video coding schemes. Second, is the proposal of a model-based approach that takes into consideration the human visual system in the analysis/synthesis parts of the video coding stages. This has been considered when designing a 3-D scene model of a person's face. Third is the proposal of fractal-based coding schemes which have been object of extensive research in the last years. Finally, the proposal of new paradigms in video coding taking into account the human visual system. In this paper, segmentation-based video coding schemes will be only considered. For the other topics, the reader is referred to[34] . A summary of the current image and video coding schemes of the first and second generation is presented in Table 1 which has been reproduced from[1] . In this table, however, the reader will notice that segmentation-based schemes are considered a particular case of two-dimensional (2-D) model approaches. Segmentation-based schemes have been the base of the MORPHECO project.

As a last comment regarding the influence of the human visual system, there is the important issue of defining distortion measures in the context of second generation. If the human visual system is so important in second generation video coding schemes, one may wonder which is the best measure to be used to evaluate the performance of the video coding system. If the mean square distortion measure used in rate-distortion theory is used, then we may incur the error of evaluating a human visual system-based coder with tools that have nothing to do with the HVS. A possible answer is suggested by Li and Forchheimer[19] the Kantorovich metric is proposed to evaluate the performance of compressed images. Although results are only available for fractal image coding, this metric is a promising measure to be used in second generation video coding schemes.

4.2 Segmentation-based video coding

The MORPHECO project has been focused on segmentation-based video coding techniques. Several approaches and considerations have been taken. In the following, we make a fast review of the main ideas contained in second generation segmentation-based video coding and then concentrate in the most important conclusions drawn in the MORPHECO project.

The first still image coding approach that was presented as a result of the properties of the human visual system

Table 1: Image coding schemes and their associated image source models

Image Source Models		**Coding Schemes**
Segmentation model	**Motion model**	
Pixel	—	PCM
Statistically dependent pixels		
Block	2-D translation	MC-DCT etc.
2-D model-based approaches		
2-D features : edges, contours, 2-D rigid regions, 2-D flexible regions, deformable triangle blocks, etc.	translation, bilinear transform affine transform, etc.	contour-based coding region-based coding object-based coding 2-D deformable triangle-based coding
3-D model-based approaches		
3-D global surface model: planes or geometric surfaces parameterized 3-D model	3-D global motion 3-D local deformation	object based coding layered representation 3-D model-based coding
Hybrid and model-assisted approaches		
combination of the above	combination of the above	2-D/3-D hybrid 3-D and MC-DCT model-assisted coding

studied in Section 4.1 was based on a contour/texture representation model of the image[17] . The decomposition and posterior coding of a still image in contours and textures does not represent, in principle, a theoretical problem. A good segmentation technique able to decompose the image in homogeneous entities called regions or objects has to be found. Then, the resulting contours and textures are coded. The basic idea behind this approach is that the contours correspond, as much as possible, to those of the entities defining the image, that is, the objects. The situation in video sequences is very different. To start with, the introduction of image motion poses the following questions:

- Must the segmentation be done in the spatial domain (spatially-based segmentation) or in the temporal domain (motion-based segmentation) or in both simultaneously?
- How is the concept of contour defined in still images extended to video sequences?
- Is it as efficient to use motion compensation as in first generation techniques?

To obtain a contour/texture representation the image sequence has also to be segmented. The result of the segmentation process gives a set of connected regions, called the partition sequence. We note in passing, that the partition sequence has been sometimes called the contour sequence as in still images. The partition is represented by the shape of the regions and their evolution in the time domain. These two informations have to be coded, as well as the interior of the regions, called also sometimes texture. Different performance, image quality and compression ratios are obtained as a function of the ability of the segmentation technique to provide homogeneous regions and the compression capabilities of the partition and texture coding techniques used.

4.2.1 Segmentation of video sequences

The segmentation can be done in the spatial domain, in the temporal domain or in both simultaneously. In the first case different criteria can be used to define homogeneous spatial entities such as contrast, size, etc. In the second case the motion information is what is used as homogeneity criterion. Some schemes segment both domains simultaneously. The election of the homogeneity criteria is conceptually difficult, as different criteria give different segmentations, which affect the performance of the overall coding system.

In the case of spatial segmentation three different approaches have been considered in the MORPHECO project. The first one is a pure intra-frame procedure where the temporal information is not taken into account. The second considers the image sequence as a 3-D signal (space plus time) and the third one is based on a recursive segmentation of the image sequence. After a detailed description of each approach, it was concluded in the MORPHECO project that the recursive segmentation is the most appropriate due to requirements of time coherence and to avoid random fluctuations of the partition sequence. Of special importance in video coding applications are the considerations of bit rate regulation and the time delay introduced in the segmentation process. Following the election of the time recursive approach, a general structure for segmentation has been proposed. In order to improve the quality of the segmentation, the partition and texture coding stages are included in the segmentation process. An interesting side result of this proposal is a hierarchical representation of the image very well suited for coding applications. For more details, the reader is referred to[20] .

Although MORPHECO has not considered temporal segmentation, it has to be accepted that in many situations what gives coherence to moving objects is precisely their motion. It does make sense then, to consider a temporal segmentation. The temporal segmentation is very intimately linked to the problem of motion estimation. For details on this topic, the reader is referred to[4] .

It is very important to remark that a lot of effort has been made in motion estimation in the field of image analysis and there exists many methods to estimate motion. But in that field the amount of overhead information

that has to be sent to the receiver it is not of concern. However, in video coding this information is fundamental and puts constraints and restrictions that may limit the quality of the motion estimation. Object-based analysis-synthesis codecs relying on temporal segmentation have been proposed with good results[24,5] , although have been applied only to sequences with moderate motion.

As a last comment on motion estimation it is fair to say that although segmentation-based video coding techniques rely on the concept of arbitrarily shaped regions, the motion estimation approaches presented in the literature up to date rely mainly at one stage of the process or the other on block matching techniques. One of the main reasons for this is the easy implementation of block matching algorithms and the very simple motion model assumed. If good motion estimation algorithms are designed in the future based totally in arbitrarily shaped regions, a big improvement in segmentation-based video coding schemes may be expected. As a good example of sharing the advantages of video coding schemes based in a hybrid approach of the first generation and an affine motion model based in regions the reader is referred to[14] .

4.2.2 Coding of the partition sequence

The resulting partition sequence of the segmentation stage has to be coded. The MORPHECO approaches followed in the coding of the partition sequence have been presented in[31,30] . Two modes of operation have to be distinguished: intra-frame and inter-frame. In the intra-frame mode only the shape and the positions of the regions have to be coded. Lossless and lossy approaches have been taken. The main lossless techniques may be divided into contour-oriented and shape-oriented. Among the first, chain code techniques have been widely used. Among the second, the morphological skeleton and the quadtree techniques have also been proposed. It has been concluded in MORPHECO that the best technique for intra-frame lossless partition coding is the chain code. Lossy techniques for the intra-frame mode have also been proposed. The amount of losses is very critical as we have already mentioned that the human visual system is very sensitive to the contours of the image. The so-called multigrid chain code has shown very promising results in a lossy scheme[30] .

In the inter-frame mode, in addition to the coding of the shape and the position of the regions, the labels of each region have also to be transmitted. MORPHECO has presented a general motion compensation strategy for the partition and has been concluded that with this scheme it is possible to divide by two the cost of the inter-frame mode with respect to the intra-frame mode.

4.2.3 Texture coding techniques

We have already explained in Section 4.1 that texture contributes to improve the quality of second generation video coding techniques (as it does also for first generation). It is important then, to provide texture representations that may help the coding of video sequences in second generation schemes. It is fair to say that practically all texture coding approaches in the context of segmentation-based video schemes are based in first generation image coding, that is, waveform coding techniques. It is clear that the main effort has been dedicated to adapting the already existing coding techniques to the arbitrary shape of the regions of the images. Very few works report the specific coding of the prediction error image, although most of them assume that the same technique used in intra-frame coding can be applied in inter-frame coding.

MORPHECO has reviewed the main texture coding techniques that have been used in segmentation-based image and video coding schemes. Special emphasis has been put on a generalized transform coding algorithm that allows for coding the texture inside image regions of arbitrary shape[6] . This approach has been generally used in most of the schemes presented so far in the context of segmentation-based image and video coding. It has the additional advantage of being able to be used in the coding of the prediction error in motion compensation video schemes. Approaches relying upon stochastic vector quantization[7] have also been proposed.

5 THE MORPHECO SCHEME

We have very briefly reviewed so far the main concepts in which the MORPHECO scheme has been founded. This section is dedicated to show some results. The objective is to show the quality that can be provided in this type of approaches when used in a very low bit-rate context. The scheme relies on spatial segmentation using a Mathematical Morphology approach. For details the reader is referred to[31] .

The general structure of the MORPHECO video coding scheme is presented in Figure 1. The scheme involves a time-recursive segmentation relying on the pixels homogeneity, a region- based motion estimation, and motion compensated contour and texture coding. The segmentation step follows a implementation based in Mathematical Morphology. The partition coding is based in the motion compensation of partitions. The texture is coded using texture compensation followed by the coding of the prediction error using a generalized orthogonal transform technique. A translational motion model is used to find the motion associated to each region. One of the important features of the approach is that no assumption is made about the sequence content. Moreover, the algorithm structure leads to a scalable coding process giving various levels of quality and bit rates. The coding as well as the segmentation are controlled to regulate the bit stream.

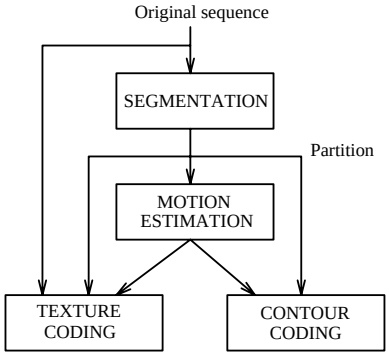

Figure 1: General structure of the MORPHECO segmentation-based coder

The algorithm has been designed to meet the following basic system requirements:

- The codec is mainly devoted to very low video bit-rates, that is below 64 kbits/s.
- It should be generic with respect to the input sequences. In particular, no assumption about the scene content should be made.
- It should be flexible with respect to the coded sequence quality. The coding strategy should provide an easy way to define various levels of quality.
- The system is principally devoted to fixed-rate transmission. As a consequence, the bit stream should be regulated.
- It should be suitable for interactive applications. Therefore, large processing delays should be avoided.

Figure 2 presents the sequence *Foreman* (frames 10 and 110) coded at 32 kbits/s at a frame rate of 5 Hz. This sequence is representative of the Class B sequences proposed in MPEG-4.

Figure 2: Above: original sequence *Foreman*. Below: coded frames at 32 kbits/s. (MORPHECO)

The general results of the MORPHECO project have shown a promising way of encoding video sequences at very low bit-rates using segmentation-based techniques. It is clear that the quality of the reconstructed images is still susceptible of improvement. The basis in which the MORPHECO scheme can be improved revolves around one very basic concept: is there a way to optimally distribute the available bits needed to encode the partition? In other words, is it possible to select each time the most appropiate way to encode a region? These considerations have materialized in the SESAME proposal[3] . As the basis of optimal bit allocation is founded on the concepts of *dynamic coding*, we find convenient to further explain this topic in the context of segmentation-based schemes in a different section, before entering into the description of the SESAME proposal.

6 DYNAMIC CODING

The first reference of bit allocation for completely arbitrary inputs and discrete quantizer sets was given by[33] . Then, the extension for more general temporally and spatially dependent coding scenarios was addressed in[28] . Finally, the application in the framework of segmentation-based coding was given in[29] . One of the major features in real world images and image sequences is the *non-stationary* behaviour of the signal. Data representing visual information in one area of an image have different characteristics than those representing another area of the same image. To take these variations into account, compression techniques need to be as adaptive as possible. It turns out that, a single method with the best possible adaptivity cannot perform as well as a team of methods each giving its best for the most appropriate part of the picture. In the original dynamic coding concept, several quantizers are used in parallel to obtain the best quantization possible. This idea can be naturally extended if several segmentation methods are used in parallel to obtain the best image segments, having always in mind that these segments have to be encoded. The ultimate goal in segmentation is to obtain segments that represent real objects present in the scene (or part of these objects). Since there is no a unique segmentation method performing very well in all situations, these results need to be *merged* in a collaborative way to yield the final result. Dynamic coding becomes then an effective competition between several compresion techniques for representing image data segments in the most efficient way. The image data are represented as the union of several regions, each approximated by a representation model most appropriate for this region. Data concerning a region are stored and/or transmitted along with their compression algorithm or their index if the receiver has

the same toolbok. Dynamic coding offers attractive features such as genericity, flexibility and openness. It is a fairly general scheme which may be used in many situations. In particular, it has been used in the SESAME proposal for MPEG-4

7 THE MORPHECO EVOLUTION: THE SESAME PROPOSAL FOR MPEG-4

The SESAME proposal was designed with the following objectives in mind:

- The algorithm does not make any assumption about the scene content. No a priori information is assumed.
- The scene can have an arbitrary number of objects with arbitrary relations, positions and motions.
- The algorithm relies on the same coding strategy to deal with low or high bit rates.
- The algorithm structure shoud be flexible enough to allow integration and comparison of new tools.

7.1 Principle and structure of SESAME

The objectives summarized above lead to three very important consequences in the definition of the SESAME proposal:

- The partition should be signal dependent. Therefore it results from an analysis of the sequence. In particular, the approach discards a priori defined partitions such as block partitions.
- The representation of objects by partitions does not only involve the definition of object contours at one instant but also their time evolution. Indeed, one should be able to recognize that one region (or one object) proceeds from a given region (or object) in the previous frames. In other words, one should be able to track regions and objects in time. This point is mandatory to be able to define content-based functionalities. This approach discards all techniques that define partitions independently from one frame to another one.
- As a consequence of the generic approach, the strategy cannot rely on a fixed topology of the partition. The partition has to evolve with the modifications of the scene content: regions are to be introduced in the partition when new objects appear in the scene. Regions are to be removed when objects disappear in the scene.

The general structure of the SESAME scheme is presented in Figure 3. The encoding process relies on three sets of functions: Partition functions, Bit allocation function and Coding functions.

- **Partition functions** As discussed before, the sequence representation relies on signal-dependent partitions. Moreover, following the sequence evolution, regions should be tracked and the partition topology may be modified. This set of requirements is implemented by the *partition functions*. In fact, two processing steps can be distinguished (see Figure 3): the *Projection* which tracks the time evolution of the regions, and the *Partition* tree which deals with the modifications of the partition topology (elimination and introduction of regions).

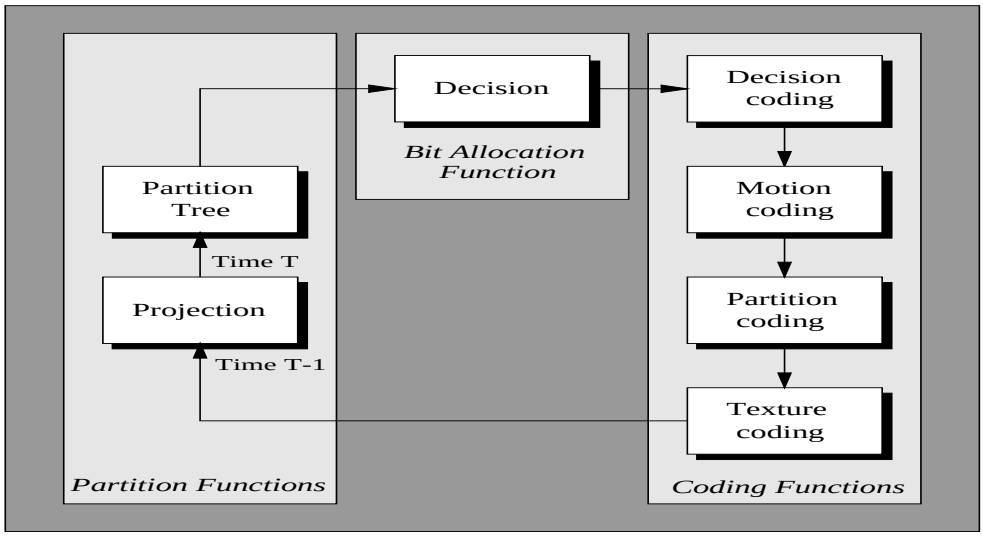

Figure 3: General structure of the SESAME scheme

- **Bit allocation function** This function is implemented by the block called *Decision*. In order to get an efficient content- based representation, the problem of bit allocation has been carefully studied. The *Bit allocation function* optimizes the repartition between the varios types of information to be coded and transmitted. In the SESAME proposal, it concerns mainly motion, partition, gray level and color information. As a result, the *Decision block* defines the coding strategy, that is, the region to be coded and the type of coding to be applied on each region. The *Decision block* has to select the best strategy in terms of regions and coding techniques among a set of possibilities. The *Decision* is made based on Rate-Distortion theory concepts.

- **Coding functions** The last set of functions actually codes the information necessary to restore the sequence on the receiver side. They deal with the encoding of the coding strategy (Decision coding), the motion information (Motion coding), the partition (Partition coding) and the gray and color level pixel values (Texture). The partition and the texture should be motion compensated. This explains why the Motion coding block is located before the Partition and texture coding blocks.

7.2 SESAME RESULTS

Figure 4 presents the sequence *Foreman* (frame 150) coded at 33 kbits/s at a frame rate of 5 Hz. It can be generally stated that the SESAME coded images present some improvements with respect to the previous MORPHECO results. In addition, the same video sequence when coded with the current standard H.263[13] , presents better quality at the same, or lower, bit-rates. However, the SESAME proposal has opened new possibilities that deserve further research. First is the introduction of the *Decision* concept that can improve in the future the bit assignment used. Secondly, is the general object-based approach taken that facilitates the introduction of new functionalities such as content-based interactivity. Finally is the introduction of different coding tools that can be used in the actual MPEG-4 Video Verification Model[9] .

Figure 4: Left: original sequence *Foreman*. Right: coded frame at 33 kbits/s. (SESAME)

8 THE FUTURE MPEG-4 STANDARD

In the last years, and as already mentioned, very significant developments happened in the field of audio-visual communications, notably in the area of video coding. The developments led to the standardisation of a number of international video coding schemes, such as ITU-T H.261 and H.263, and ISO/IEC MPEG-1 and MPEG-2, addressing a large range of applications with different requirements, e.g. in terms of bit-rate, quality or delay. The audio-visual services nowadays available, mainly provide the ability to see (and hear from) places and times where we have never been. In fact, the world arrives to us in the form of a sequence of 2D frames, coded exploiting the statistical characteristics of the luminance and chrominance signals. However, the capability of *vision* is just a part of the question. Typically, the human being needs and wants to see, to take actions after, interacting with the objects that compose the world being seen[25] . MPEG-4 aims to provide a universal, efficient coding of different forms of audio-visual data, called audio-visual objects. This basically means that MPEG-4 intends to represent the world understood as a composition of audio-visual objects, following a script that describes their spatial and temporal relationship. This type of representation should provide the possibility that the user interacts with the various audio-visual objects in the scene, in a way similar to the actions taken in everyday life. Although this content-based approach to the scene representation may be considered *evident* for a human being, it represents in fact a revolution in terms of video representation architecture used in the available standards, since it allows a *jump* in the type of functionalities that may be provided to the user. A scene represented as a composition of (more or less independent) audio- visual objects offers to the user the possibility to *play* with the scene content, by changing some of the objects characteristics (e.g. position, motion, texture or shape), by accessing only selected parts of scene or even by cut and pasting objects from one scene to another. Content and interaction are thus central concepts in MPEG-4.

Another clear limitation of the available audio-visual coding standards is the consideration of a restricted number of audio and video data types. MPEG-4 wants to consider and harmoniously integrate natural and synthetic audio-visual objects, including mono, stereo and multi-channel audio, as well as either 2D and 3D, and either mono, stereo or multiview video. This integration course should be extended also to the audio-video relation to exploit the mutual influence and interdependence between these two types of information. Finally, the integration course is to be applied to the analysis and coding tools used since new and already available tools will be integrated with the only target to reach the best possible content-based standard.

The rapidly evolving technological environment of the last years showed in a clear way that standards which do not take into account the continuous development of the hardware and of the methodologies and just want to fix a solution, risk to become obsolete relatively soon. The last main direction underlying MPEG-4 is thus flexibility and extensibility. These features are essential in the current moving technological landscape and should be provided by a syntactic description language called *MPEG-4 Syntactic Description Language (MSDL)*. The MSDL will provide the extensibility not only to build new algorithms by selecting and linking pre-defined tools (level 1), but also by *learning* new tools downloaded by the encoder[10,11] . In the definition

of the official MPEG-4 focus, the three main driving forces mentioned above - content and interaction, integration and, flexibility and extensibility - have been matched in a vision of the technological world, where the convergence between the telecommunications, computer and TV/film areas is growing, leading to the mutual exchange of elements, formerly typical for each one of these areas[16] .

8.1 The MPEG-4 new or improved functionalities

As happened for the already available MPEG standards, MPEG-4 does not want to address any specific application but prefers to support as many clusters of functionalities which may be useful for various applications as possible. A functionality is here understood as an application capability that users may choose to use or not depending on their needs. This functionality-based strategy is represented in MPEG-4 by the eight *new or improved functionalities*, described in the MPEG-4 Proposal Package Description[16] . These functionalities were considered useful in the context of the three converging worlds above mentioned and they are not nowadays conveniently addressed by the available or emerging standards. The eight new or improved functionalities have been clustered in three sets - content-based interactivity, compression and universal accessibility - depending on which one they primarily address. Note that the three sets of functionalities are not orthogonal and that a functionality may well contain characteristics of a set in which it was not classified. The MPEG-4 new or improved functionalities are:

- **Content-based interactivity**
  - Content-based multimedia data access tools
  - Content-based manipulation and bitstream editing
  - Hybrid natural and synthetic data coding
  - Improved temporal random access
- **Compression**
  - Improved coding efficiency
  - Coding of multiple concurrent data streams
- **Universal access**
  - Robustness in error-prone environments
  - Content-based scalability

The current set of new or improved functionalities resulted as a compromise between the various sentiments present in MPEG at the time of its definition. These functionalities are not all equally important, neither in terms of the technical advances they promise, nor the application possibilities they open. Moreover, they seem to imply a few rather ambitious goals which, although important and in line with the MPEG-4 vision, can only be reached in due time, and provided that a proper terminal architecture be available.

8.2 The MPEG-4 video verification model

As happened in the past for MPEG-1 and MPEG-2, MPEG issued, in July 1995, a call for proposals of audio and video tools and algorithms, in order to gather the technical information necessary for the achievement of its targets. These tools and algorithms have been evaluated in November 1995 and January 1996. In the context of MPEG-4, a tool is a technique that is accessible via the MSDL or described using the MSDL. An algorithm is an organized collection of tools that provides one or more functionalities. A profile is an algorithm or a combination of algorithms, constrained in a specific way to address a particular class of applications. At the MPEG-4 video tests, the bit-rates ranged from 10 to 1024 kbit/s[26] . The video test material was divided in 5 classes of sequences, 3 of which clearly address low or very low bit-rates: class A - low spatial detail and low amount of motion - 10, 24 and 48 kbit/s; class B - medium spatial detail and low amount of motion or vice-versa - 24, 48 and 112 kbit/s; class C - high spatial detail and medium amount of motion or vice- versa - 320, 512 and 1024 kbit/s; class D - stereoscopic sequences (no formal subjective testing was performed) and class E - hybrid natural and synthetic content - 48, 112 and 320

kbit/s. Some tens of tools and algorithms were presented to the first MPEG-4 evaluation phase. Following the results of the formal video subjective tests performed for three representative functionalities - content-based scalability, compression and error robustness[27] , two main conclusions may be driven: i) conventional block-based hybrid (DCT/motion compensation) schemes still perform very well in terms of compression, for the whole range of sequences and bit-rates tested and, ii) the provision of content-based functionalities mainly depends on the data structure used; this means that, provided the adequate data structure is used, almost any type of coding tools may be used (at least in principle). In this context, at the Munich meeting (January 1996), the MPEG video group defined the common platform to start the MPEG collaborative phase[9] . The collaborative phase is one of the main strengths of the MPEG work since all the experts are asked to improve and optimise the same codec. In MPEG-4, the common platform is known as the verification model (VM). The VM is a completely defined encoding and decoding environment such that an experiment performed by multiple independent parties will produce essentially identical results[26] . New tools can be integrated in the VM, substituting other tools, when the corresponding core experiment has shown significant advantages in this integration.

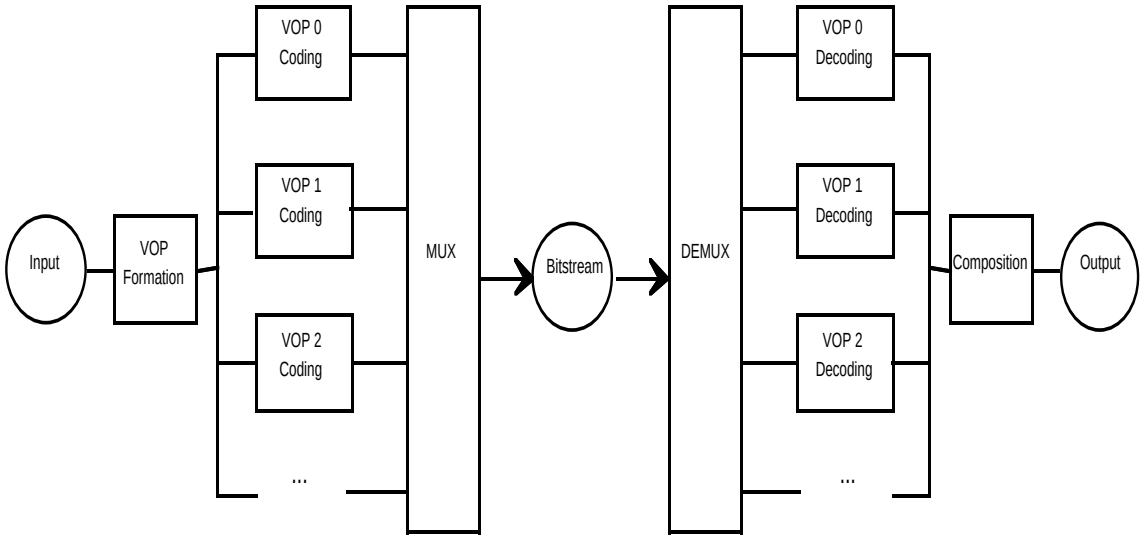

Figure 5: MPEG-4 Video VM encoder and decoder architectures

The Video VM defined in Munich addresses only arbitrary shaped 2D objects and represents the scene as a composition of these objects, now called Video Object Planes (VOPs). This VM is shown in Figure 5. Each VOP is a video accessible unit and should, in principle, be coded independently of the other VOPs, although for coding efficiency reasons some dependence may be considered in the future. The scene is divided in VOPs either automatically, semi-automatically or fully manually or, the VOPs may be initially available if the scene is created as a composition of material taken from different sources. Since MPEG-4 wants to be able to represent any type of VOP composition, independently of the way it has been obtained, the VOP formation block is not considered in the MPEG-4 Video VM. This means that the input is already in the form of four components - one luminance, two chrominances and an alpha-plane representing the blending contribution of each VOP for every part of the scene. The VOPs may use different spatio-temporal resolutions depending on their own intrinsic characteristics; this is in line with the MPEG-4 basic targets, having the content driving the representation and coding. Although MPEG-4 will mainly standardise the representation architecture and the coding tools, some analysis tools may integrate the standard since, for many applications, its success will also depend on the easy availability of this kind of tools. In terms of coding and following the conclusions of the MPEG-4 tests already mentioned, each VOP is coded using the tools present in the ITU-T H.263 coding standard. The main differences are related to the transmission of the alpha-plane (shape) information for each VOP and to the possibility that motion and texture information are separate at the VOP level. The macroblocks falling over the VOP contour are filled using a padding

technique. The choice of block-based hybrid schemes to code the texture of each MPEG-4 VOP (at least in a first approach) highlights again the high performance of this texture coding approach, already largely used in second generation schemes The syntax defined for the MPEG-4 Video VM follows very closely the approach used to define the representation architecture - a scene is a composition of VOPs[9] . In this context, only two bitstream layers have been defined: i) the session layer, which encompasses a given span of time, contains all the video information needed to represent this span of time, without reference to other session layers and, ii) the VOP layer contains the syntactic elements related to each VOP, notably the identifier, the temporal reference, the spatial reference, the width and height, the visibility (displayed or not), the composition order and a scaling factor to be used during the composition process. This first MPEG-4 Video VM will very likely be significantly improved during the next months, leading to a first MPEG-4 Video Working Draft (WD) by November 1996.

8.2.1 First, second and ... third generation video coding techniques

A careful analysis of the so called first and second generation video coding techniques allows to conclude that the main difference between these two generations of coding techniques is the way they understand and organise the data corresponding to the world they intend to represent and the role and relevance in this process of the human user. In the sequel we will use the term second generation video coding techniques mainly in the context of the particular case of segmentation-based techniques.

While first generation coding techniques basically organise the world in square blocks, second generation techniques, although sometimes using square blocks to compensate the motion or code the texture, use as basic units, arbitrary-shaped regions. In fact, the second generation techniques want to take into account two facts: i) the world is composed of objects, basically defined by their contours to which the human being is particularly sensitive (see Section 4.1) and, ii) spatial and temporal redundancies are related to objects in the scene with arbitrary shape and thus redundancy reduction may be best done by considering arbitrary shaped regions as the basic coding units (even if square blocks are sometimes and somehow used to code these units). Of course, the understanding of the world as a composition of objects is a big step in the direction of allowing the user to play with the scene content even if second generation techniques did not usually offer the independent access to each object in the bitstream.

In this context, and looking to Figure 5 where the MPEG-4 VM codec structure is presented, it may be said that MPEG-4 is closer to the second generation than to the first generation techniques in the sense that also MPEG-4 understands the scene as a composition of arbitrary shaped objects (VOPs in the VM). This basically means that MPEG-4 and second generation coding schemes are close in the way they organise the video data taking into account that objects are relevant for human beings because of the relevance of contour information, because of content-based functionalities or whatever. At the same time, the MPEG-4 VM uses (at least in its first version of January 1996) typical first generation coding techniques to code the texture of each VOP, as already happened for many second generation coding schemes. However MPEG-4 goes beyond first and second generation coding techniques because it got free of the conventional 2D frames, which is a constraint present both in first and second generation coding schemes. In fact, MPEG-4 is able to represent not only scenes with only one object or VOP - situation close to the first generation schemes - or scenes with two or more objects or VOPs mutually disjoint, resulting from the segmentation of a 2D scene - situation closer to the first generation schemes, but also scenes with two or more objects resulting from a composition from several sources independently available, 2D or even 3D. In a similar way second generation schemes used the best of first generation schemes, we are now facing the birth of third generation schemes exploiting and exceeding the best of both first and second generation techniques. While the concept of segmentation is intrinsic to the analysis part of the second generation coding schemes, with the target to define the objects in the scene which is always a sequence of 2D frames, the situation in MPEG-4 is more complex since it has to consider the same situation and additionally other situations. To clarify this question, it is convenient to define two types of segmentation:

- **Segmentation for representation** This segmentation is necessary for the object/VOP definition, when starting from a sequence of conventional 2D video frames. It basically follows semantic criteria

corresponding to the application in question. Segmentation should lead to a set of meaningful VOPs which are subsequently available for access and manipulation. This is the type of segmentation intrinsically present in second generation coding schemes, particularly in segmentation-based approaches, although the segmentation criteria were typically more based on homogeneity than on semantics.

– **Segmentation for coding** This segmentation is made for compression purposes and thus each VOP is divided in a set of regions which are considered homogeneous in some way, and for which the spatial and temporal redundancy will be reduced through prediction. Mainly compression efficiency criteria apply, as the regions are not meant to be used for any user controlled interaction.

Note that the two types of segmentation may either coexist or be used independently. This means three cases are possible: i) the VOPs (accessible units) are defined through an automatic segmentation and, after that, each VOP is coded using an arbitrary shape region-based approach for the motion or for the texture (a VOP may still include several regions with different homogeneity); ii) the VOPs are explicitly available (no need to do segmentation for representation) or there is only one VOP and the coding is basically arbitrary shape region-based (redundancy is exploited using arbitrary shaped regions); iii) the VOPs are defined through segmentation of 2D video frames but the coding is not region-based (e.g. redundancy is exploited using square shaped regions). In second generation coding schemes, the two types of segmentation above mentioned were basically coincident in the sense that the objects identified were also used for coding, since as they were often identified following homogeneity criteria this allowed to foresee that they would also be good redundancy reduction units. Unlike in MPEG-4, most of the second generation coding techniques did not code the partition sequence in a way that objects could be independently accessed in the bitstream. Although the (more or less) independent coding of objects has a price in terms of coding efficiency, it is essential to provide many of the new MPEG-4 functionalities, notably content-based scalability and manipulation. In MPEG-4, the segmentation for representation will be needed or not depending on the application and on the point in the production process where the source information is available. Also, when segmentation for representation is needed, the specific method to be used, either automatic, semi-automatic or manual will depend on the application and its requirements, e.g. on delay. Note that, in MPEG-4, it is expected that the number of objects or VOPs to be defined basically depends on the application and its semantic criteria and will not be too high (the current VM uses a maximum of 32). This means that we will very likely have VOPs that are formed by several homogeneous areas and thus the same techniques used in second generation coding techniques, including their segmentation (now segmentation for coding), may now be used in the context of a VOP, if they prove to be more efficient that first generation techniques (at least in some conditions).

9 CONCLUSIONS AND FUTURE TRENDS

This paper has reviewed the main concepts in which second generation image and video coding techniques are founded along with an explanation of the current state of the art in MPEG-4 standardization activities. Second generation approaches have been introduced as a way to overcome the main conceptual problems posed by first generation and to facilitate the opening to the new world of functionalities. Second generation relies on an understanding of the image as a composition of objects which have a semantic meaning. The MORPHECO project and the SESAME proposal have been presented as examples of segmentation-based schemes. These schemes have been presented as representatives of a new generation of codecs that, taking into account the properties of the human visual system, pay attention to the content of the image. New functionalities have been shown as opening a very new and exciting field of applications.

Now it is time to ask which is the next step. As usual this question has a very difficult answer. It is clear that the next step in segmentation-based coding is not to segment objects but to recognize them. If we were able to recognize objects, then everything, motion compensation, object tracking, etc., would be easier. In a more conservative context we will see an explosion of hybrid first and second generation schemes taking into account the good properties of both. In addition, combined second generation source and channel video coding should provide a better efficiency.

Speech may come to the rescue of video (or vice versa) by combining in an intelligent way both fields. In addition promising avenues to pursue are the further involvement of the human visual system in the designing of the coding scheme, new approaches to the estimation and employment of motion along with signal-dependent coding, the re-use of the transmitted information and the designing of new coding architectures. Dynamic coding will prove very useful in the design of the new generation of codecs.

MPEG-4 activities are seen as a natural way of evolution to seek for new ways of image representation and understanding. In conclusion, it may be said that video coding is evolving by integrating, in the context of new representation architectures, more and better tools with the already available ones that keep performing well, towards the provision of new and more complex video services and functionalities.

10 REFERENCES

[1] K Aizawa. Model-based video coding. In L. Torres and M. Kunt, editors, *Video coding: the second generation approach*, pages 305–335. Kluwer Academic Publishers, 1996.

[2] R. J. Clarke. *Digital compression of still images and video.* Academic Press, 1995.

[3] I. Corset et al. Segmentation-based coding system allowing the manipulation of objects (SESAME). Technical Report ISO/IECJTC1/SC29/WG11/MPEG95/408, LEP and UPC and CMM, November 1995.

[4] F. Dufaux and F. Moscheni. Segmentation-based motion estimation for second generation video coding techniques. In L. Torres and M. Kunt, editors, *Video coding: the second generation approach*, pages 219–263. Kluwer Academic Publishers, 1996.

[5] P. Gerken. Object-based analysis-synthesis coding of image sequences at very low bit rates. *IEEE Transactions on Circuits and Systems for Video Technology*, 4(3):228–235, June 1994.

[6] M. Gilge, T. Engelhardt, and Mehlan R. Coding of arbitrarily shaped image segments based on a generalized orthogonal transform. *Signal Processing: Image Communication*, 1(2):153–180, October 1989.

[7] D. Gimeno, L. Torres, and Casas J.R. A new approach to texture coding using stochastic vector quantization. In IEEE, editor, *Image Processing Conference*, Austin (TX), USA, November 1994.

[8] B. Girod. Rate-constrained motion estimation. In *Proc. SPIE Visual Communications and Image Processing VCIP-94*, volume 2308, pages 1026–1034, Chicago, USA, September 1994.

[9] MPEG-4 Video Group. MPEG-4 video verification model 1.0 - ISO/IEC JTC1/SC29/WG11 N1172. January 1996.

[10] MSDL Ad Hoc Group. Requirements for the MPEG-4 syntactic description language - ISO/IEC JTC1/SC29/WG11 N1022. July 1995.

[11] MSDL Ad Hoc Group. MSDL specification, version 1.0 - ISO/IEC JTC1/SC29/WG11 N1164. January 1996.

[12] ITU-T Recommendation H.261. Video codec for audiovisual services at px64 kbit/s. Technical report, ITU, 1993.

[13] Draft ITU-T Recommendation H.263. Video coding for narrow telecommunication channels at less than 64 kbit/s. Technical report, ITU, July 1995.

[14] H. Jozawa. Segment-based video coding using an affine motion model. In *Proc. SPIE Visual Communications and Signal Processing VCIP-94*, volume 2308, pages 1605–1614, Chicago, USA, October 1994.

[15] ISO/IEC IS 10918-1 (JPEG). Digital compression and coding of continuous-tone still images: requirements and guidelines. Technical report, ISO, 1994.

[16] ISO/IEC JTC1/SC29/WG11. MPEG-4 Proposal Package Description (PPD). July 1995.

[17] M. Kunt, A. Ikonomopoulos, and M. Kocher. Second generation image coding techniques. *Proceedings of the IEEE*, 73(4):549–575, April 1985.

[18] C. Labit and J. P. Leduc. Very low bit rate (VLBR) coding schemes: a new algorithmic challenge? In *Proc. SPIE Visual Communication and Image Processing VCIP-94*, volume 2308, pages 25–37, Chicago, USA, September 1994.

[19] H. Li and R Forchheimer. Extended signal-theoretic techniques for very low bit-rate video coding. In L. Torres and M. Kunt, editors, *Video coding: the second generation approach*, pages 383–428. Kluwer Academic Publishers, 1996.

[20] F. Marqués, P. Salembier, and M. Pardàs. Coding-oriented segmentation of video sequences. In L. Torres and M. Kunt, editors, *Video coding: the second generation approach*, pages 79–123. Kluwer Academic Publishers, 1996.

[21] ISO-IEC IS 11172 (MPEG-1). Coding of moving pictures and associated audio for digital storage media up to about 1.5 Mbit/s. Technical report, Motion Picture Experts Group, 1993.

[22] ISO-IEC DIS 13818 (MPEG-2). Generic coding of moving pictures and associated audio. ITU-T recommendation H.262. Technical report, Motion Picture Experts Group, March 1994.

[23] H. Musmann, P. Pirsch, and H. J. Grallert. Advances in picture coding. *Proceedings of the IEEE*, 73(4):523–549, April 1985.

[24] H.G. Musmann, M. HOtter, and J. Ostermann. Object-oriented analysis-synthesis coding of moving images. *Signal Processing: Image Communication*, 1(2):117–138, October 1989.

[25] F. Pereira. MPEG-4: a new challenge for the representation of audio-visual information. In *Picture Coding Symposium*, Melbourne, Australia, March 1996.

[26] F. Pereira (editor). MPEG-4 testing and evaluation procedures document - ISO/IEC JTC1/SC29/WG11 N999. July 1995.

[27] H. Peterson (editor). Report of the ad hoc group on MPEG-4 video testing logistics - ISO/IEC JTC1/SC29/WG11 N999. November 1995.

[28] K. Ramchandran, A. Ortega, and M. Vetterli. Bit allocation for dependent quantization with applications to multiresolution and mpeg video coders. *IEEE Transactions on Image Processing*, 3(5):533–545, September 1994.

[29] E. Reusens. Joint optimization of representation model and frame segmentation for generic video compression. *EURASIP Signal Processing*, 46(11):105–117, September 1995.

[30] P. Salembier, F. Marqués, and A. Gasull. Coding of partition sequences. In L. Torres and M. Kunt, editors, *Video coding: the second generation approach*, pages 125–169. Kluwer Academic Publishers, 1996.

[31] P. Salembier, L. Torres, F. Meyer, and C. Gu. Region-based video coding using mathematical morphology. *Proceedings of the IEEE*, 83(6):843–857, June 1995.

[32] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27; Part I and II:379–423; 623–656, 1948.

[33] Y. Shoham and A. Gersho. Efficient bit allocation for an arbitrary set of quantizers. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 36:1445–1453, September 1988.

[34] L. Torres and M. Kunt. *Video coding: the second generation approach.* Kluwer Academic Publishers, Englewood Cliffs, 1996.